DOCUMENT RESUME

ED 061 651 EA 004 272

AUTHOR Overturf, Leonard O.; Fastman, Jerry
TITLE Student Generated Section, Course and Alternate Requests as the Keystone of a Computer Based, Student Responsive, Advance Registration and Scheduling System. College and University Machine Records Annual Conference. (17th, Columbus, Ohio, May 1972.)
INSTITUTION Ohio State Univ., Columbus.
PUB DATE May 72
NOTE 14p.

EDRS PRICE MF-$0.65 HC-$3.29
DESCRIPTORS College Administration; *Computer Oriented Programs; Enrollment Projections; Information Needs; *Management Information Systems; Models; *Scheduling; *School Registration; *Student Reaction; Student Records; University Administration

ABSTRACT

This document describes a system of computerized advance registration at Colorado State University. The main objective is to generate a schedule that will satisfy student requests and needs, given constraints of faculty, time, and space. The system consists of a series of computer programs and a set of well-documented manual procedures in the areas of course schedule construction and maintenance, student-generated course/section request processing, student scheduling, and add/drop processing. (RA)

ED 061651

Student Generated Section, Course and Alternate Requests as the Keystone of a Computer Based, Student Responsive, Advance Registration and Scheduling System

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
OFFICE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL OFFICE OF EDUCATION POSITION OR POLICY.

LEONARD L. OVERTURF AND JERRY FASTMAN

Colorado State University

MANAGEMENT INFORMATION 17th annual CUMRC INFORMATION MANAGEMENT

College and University Machine Records Conference May 1972

THE OHIO STATE UNIVERSITY COLUMBUS, OHIO

EA 004 272

STUDENT GENERATED SECTION, COURSE AND ALTERNATE REQUESTS AS THE KEYSTONE OF A COMPUTER BASED, STUDENT RESPONSIVE, ADVANCE REGISTRATION AND SCHEDULING SYSTEM

Mr. Leonard L. Overturf
Director, Admissions and Records
Colorado State University

Mr. Gerald E. Fastman
Principal Consultant
Systems and Computer
Technology Corporation (SCT)

The Problem

Perhaps the most frustrating, time-consuming and costly process in the administration of a college or university today is the registration process. Students become frustrated by long lines, course closings and lack of sufficient time to select courses. Administrators become perplexed about the drop/add situation, low student enrollment in many classes, the clerical tasks involved in registration and the time and money expended in scheduling students.

The basic design of the former Colorado State University student registration and scheduling system evolved over the past several years as the demands of a rapidly expanding university increased. Refinements made through the '60's made it possible for the system to continue to serve the University as its student body increased to above 17,000 students.

The rapid growth of the '60's - increasing not only the size but also the complexity of the University - had placed a growing burden on the existing student registration and scheduling system to the point where the University had to make some realistic decisions as to the steps necessary to better serve the students, faculty, staff and administration in the most effective way. The registration procedure depended primarily on the shuffling and reshuffling of punch cards by a crew of part-time employees over a two to three week period supplemented by unit-record and computer equipment. As a result, prompt response to unexpected changes in the pattern of student course requests, proper control of class size, necessary flexibility in meeting the problems of individual students, and even the prompt, accurate recording of total enrollment figures were becoming more difficult to achieve.

At Colorado State University, as in most institutions of higher learning, curricula are becoming less structured and a wider variety of courses is now offered. There has also been an increase in the number of single and double section courses. These factors have made the construction of the master schedule of courses a complex affair. Additionally, as the curricula have become less structured and as patterns of student life have changed, it has

become more difficult to make accurate predictions about student course demand patterns. As a result, departments need more information about changing demand as well as more time prior to submission of course offerings. Also needed is the capability of storing and displaying course section demand data and the mismatch with the master schedule in a form that will facilitate current and future allocation of faculty, time and space.

Students today seek a greater participation in the determination and selection of course offerings. The desire for more flexibility, changing student values, and the increasing numbers of students who work make it necessary to change the current practices of student scheduling.

The problem then is how can we schedule students with their own requests given the constraints of faculty, space and time, particularly as they relate to a requested time pattern.

COLORADO STATE UNIVERSITY SYSTEMS REQUIREMENTS

The "people" systems requirements of the Colorado State University registration and scheduling system have the following characteristics and capabilities:

A. Students

- Minimize the data reporting requirements levied upon students by eliminating all duplicated data collection procedures.

- Reduce the inconveniences associated with enrolling in classes by supporting an "advance" registration process requiring minimum input on the part of students.

- Provide mechanisms which enable student course and section demand data to be reported to faculty and administrators and provides procedures which enable the university to respond to that demand.

- Provide computerized scheduling of students based upon students' expressed section preferences and permit students to specify free time requests and alternate course requests.

B. Faculty

- Provide support for the clerical functions of developing and refining a schedule of classes to be offered in an academic quarter.

- Provide reports which enable past course schedules to be used as a basis for developing future course schedules and reduce the amount of data required to generate a schedule for subsequent quarters.

- Provide for fast and accurate reporting of faculty schedule and teaching load data to insure equitable distribution of teaching load over the entire faculty, and indicate areas requiring the acquisition of additional faculty.

- Provide reports which enable colleges and departments to react to student course and section demand thus enhancing the allocation of teaching resources.

- Support an "advance" registration process which minimizes hurried responses on the part of faculty to rapidly developing student course and section demand.

- Provide for specifying section content by major, including major only sections and major excluded sections.

- Promote section balancing.

C. Administration

- Provide numerous reports to support planning in the university and which promote better allocation and acquisition of resources.

- Support long term planning for faculty allocations.

- Permit students to be scheduled in one group, or several, separate scheduling runs against remaining capacity and/or an updated master schedule.

- Minimize the clerical burden and confusion associated with the registration process.

- Provide a fully documented system including both the computer portion and the associated manual procedures.

Description of Colorado State University Registration and Student Scheduling System

The Colorado State University registration and student scheduling system is based upon the "people" systems requirements outlined in the previous section. Two key concepts based on these requirements are an "advance" registration process and a "rolling" scheduling capability.

Advance registration means that students make specific section requests for the forthcoming term in the closing weeks of the current term. For example, students submit their requested enrollments for winter term in the closing weeks of fall term. An advance registration process enables faculty and administrators to respond to student course and section demand. The process provides mechanisms which report on student demand by section and enable faculty and administrators to respond to that demand by modifying the initial schedule of sections to be offered.

Advance registration begins when faculty compile an initial schedule of sections to be offered in the forthcoming term. This schedule, based on previous schedules, is compiled and refined early in the preceding term. This initial schedule is based upon knowledge of request patterns in previous terms and the availability of faculty and class rooms. Students make specific section requests for enrollments using the initial course schedule. Requests are tallied and faculty, department chairmen and administrators make modifications to the course schedule in an effort to respond to actual student section demand. The course schedule may be modified by adding sections to courses, deleting sections, or adding capacity to sections.

Rolling scheduling permits scheduling of students in discrete groups with automatic protection of cumulative scheduling results to allow for scheduling by class, late registrants, newly admitted students, or any other group of students.

The system consists of a series of computer programs and a set of well-documented manual procedures in the areas of course schedule construction and maintenance, student generated course/section request processing, student scheduling, and add/drop processing.

All system transactions are communicated to the various computer programs of the system by a common "front-end" called a Transaction Input Module. It accepts transactions for the establishment and maintenance of the schedule of course sections, student course section request sets, and academic program changes (adds and drops). The module reformats input

transactions into an internal system format, sequences the transactions into proper sequence for updating the files, and validates all transactions for proper format and content. The module rejects all transactions which fail to meet validation criteria and lists all rejected transactions. A series of control reports is also produced which list counts of valid and invalid transactions by type of transaction. The reports are also used to balance with counts produced later by other modules processing the transactions.

The Course Schedule Maintenance Module provides computer support for many of the clerical chores associated with building and maintaining a file of course offerings for use in conjunction with a computerized student scheduling system. This module builds and maintains a file containing records describing each course section to be offered in a given quarter. The records for each course section contain the title of the course, the instructor, course section, a description of each class meeting for the course section, and such other data as the predetermined number of majors per sections, to allow for reservation of seats for specific types of students.

The update program permits any field in the course schedule record to be modified. The program also permits records to be added to, or deleted from, the Course Schedule Master File. The program produces an update audit trail, which provides a hard copy record of each modification to the Course Schedule Master File. This report lists the results of every transaction and each course schedule record either added to or deleted from the file. The program also produces a control report which lists input and output file counts and counts of transactions processed by transaction type. Report records are generated, which on a parameter card basis, are used to print any combination of the following reports: Schedule Working Report, Faculty Schedule Report, Schedule of Recitations, Reference Number Report, and Room Schedule Matrix.

The Course Schedule Maintenance Module provides computer support during the creation and refining of a quarter's Course Schedule, and after scheduling a portion of the student body during rolling scheduling.

The Course Request Processing Module builds and maintains a file of student course section requests. This module preprocesses all such requests prior to submission of the requests to the Scheduling Module. Validation of the student requests is performed according to specified fixed and variable (by parameter card) validation criteria. The criteria include: number of credit hours requested, violation of Pass/Fail option, duplicate course requests, student generated time conflict, requests for oversubscribed course sections, requests for cancelled courses, requests for non-existent course or course

sections, violation of sex restriction for a course, violation of major or class restriction, partial requests for linked courses, course requests generated by students with no student master file record, and maximum number of allowable course requests exceeded.

When all validation processing has been completed, the program passes the Course Request Master File records for an individual student to an update process. This update process permits whole new sets of course requests for an individual student to be added to the Course Request Master File, and also permits individual course requests for a student to be added to the file, deleted from the file or changed. The program writes the course request records for each student resulting from this update process to a report file.

The program produces an update audit trail which provides a hard copy record of every update transaction, the entire course request set for each student whose course request set has been modified or updated, and the entire course request set for each student whose course request set contains a detectable error.

The program also produces a control report which lists file counts and transaction counts accumulated during validation and update processing. Report records are generated, which on a parameter card basis, are used to print any combination of the following reports: Section Request List Report, Section Request Tally Report, and Student Request List Report.

The Course Request Processing Module provides computer support during the collection and validating of student course section requests, and provides the information required to make the Course Schedule more consistent with student demand patterns. A registration file, which contains both the latest course schedule master file and course request master file, is the prime input to the Scheduling Module.

The Student Scheduling Module schedules students into course sections based on students' expressed section preferences. The scheduling algorithm enables student requests to be arranged in any desired priority sequence; currently, total hours earned within class. It processes student major course requests prior to other course requests. The algorithm permits the assignment of a priority to individual students; students may indicate free time requests, as well as alternates for individual course requests.

Departmental administrators establish the capacity, defined as the maximum number of students permitted in a section of a course. Capacity is

further divided into the predetermined number of majors per section (Capacity A) and the predetermined number of non-majors per section (Capacity B). The registrar establishes a balance point for each section in order to facilitate a more even distribution of major and non-major students among sections of a course. The balancing portion is that point of capacity, either Capacity A or Capacity B, which is above the desired balance point and is used to equalize section enrollments. If no balance point is established, a fail-safe balance point, determined by the registrar at the time of scheduling, is applied.

Departments are able to prescribe the number of majors permitted in each section of each course. The majors are entered into the master course schedule prior to student scheduling. The number of majors and the balance point interact independently so that CSU may establish:

No balance point and no limit
A balance point and no limit
No balance point and a limit
A balance point and a limit

The Scheduling Module uses as input the registration file produced by the Course Request Processing Module or the enrollment file from the previous scheduling run. The registration file, on magnetic tape, contains the latest version of the Course Schedule Master File and the Course Request Master File containing all of the course requests generated by students in the registration process. The registration file can be sorted such that students are scheduled according to their accumulated credit hours and/or other input sequence desired by CSU. The enrollment file contains all of the students scheduled thus far.

The program reads the registration file and extracts all Course Schedule Master File records from the registration file, reformats them and writes a record, describing each course section offered, to a disc file.

The student scheduling program begins its processing by creating a table in memory of remaining course section capacities. When this process is complete, the actual scheduling operation is ready to begin.

The student scheduling program reads all course requests for a student and places them into an internal matrix, the request matrix. After all requests for a student have been placed in the matrix, the program schedules the student by using a series of four algorithm segments, each containing from one to five scheduling steps.

The scheduling algorithms operate as follows:

A. Segment 1 - Major Requests

Step 1 - Attempts to satisfy original section preference for major course requests, against the balance point of Capacity A.

Step 2 - Attempts to satisfy major course requests by using a different section which meets at the same time as the requested section, against the balance point of Capacity A.

Step 3 - Attempts to satisfy major course requests by using a different section which meets at the same time as the requested section, against the balance point of Capacity B.

Step 4 - Attempts to satisfy major course requests by generating a section that does not conflict with the already scheduled course requests and has the most potential seats remaining, considering pending major requests, against Capacity A.

Step 5 - Attempts to satisfy major course requests by generating a section that does not conflict with the already scheduled course requests and has the most potential seats remaining, considering pending non-major requests, against Capacity B.

B. Segment 2 - Non-Major Requests

Step 1 - Attempts to satisfy original section preference for non-major course requests, against the balance point of Capacity B.

Step 2 - Attempts to satisfy non-major course requests by using a different section which meets at the same time as the requested section, against the balance point of Capacity B.

Step 3 - Attempts to satisfy non-major course requests by using a different section which meets at the same time as the requested section, against the balance point of Capacity A if major in one of the other sections.

Step 4 - Attempts to satisfy non-major course requests by generating a section that does not conflict with the already scheduled

course requests and has the most potential seats remaining, considering pending non-major requests, against Capacity B.

Step 5 - Attempts to satisfy non-major course requests by generating a section that does not conflict with the already scheduled course requests and has the most potential seats remaining, considering pending major requests, against Capacity A if major for that section.

C. Segment 3 - Course Request Scheduling

Step 1 - If a partial schedule still remains after the completion of Segments 1 and 2, the partial schedule is maintained. Processing continues, disregarding section preference, and attempts to generate a full schedule containing all original course requests, utilizing the balancing portion of Capacity A for major courses and the balancing portion of Capacity B for non-major courses. The student will receive this schedule if a full schedule can be generated.

D. Segment 4 - Student Generated Alternate Course Section Scheduling

Step 1 - If a complete schedule cannot be generated by Segment 3, the partial schedule generated in Segments 1 and 2 is restored. Processing continues, attempting to generate a full schedule, by utilizing the student alternate course section request for the remaining unsatisfied course requests, against the balance point of Capacity B.

Step 2 - Attempts to satisfy student alternate course requests by generating an alternate section that does not conflict with already scheduled courses, against the balance point of Capacity B.

After all new students have been scheduled in this run, the program searches a parameter card to determine if this is one of the succeeding runs of rolling scheduling, and if the enrollment file from the previous scheduling run is present. If the enrollment file is not present, the program passes control to the final step of the algorithm; otherwise the program begins reading the old enrollment file, one student at a time, along with all of the student's scheduled and unscheduled courses. All courses are passed through the report

writer routine to enable the system to merge all previously scheduled students into a new, updated enrollment file. This process ensures that all students will appear on the requested reports, including the Section Schedule List and Section Schedule Tally. The system provides, on option, for full and partial schedules to be printed for the most recently scheduled students only, or for all students. Optionally, seat cards are produced for use in the add/drop process.

The final phase of the student scheduling program reads the Course Schedule Master File and inserts the updated remaining section capacities in each course section record. This process occurs before writing the latest updated version of the Course Schedule Master File out to tape for the creation of the new enrollment file.

The Add/Drop Processing Module provides computer support for the maintenance of the enrollment file, from the Student Scheduling Module, with add/drop and withdrawal transactions. Validation of the student transactions is performed according to specified fixed and variable (by parameter card) validation criteria. The criteria are essentially the same as covered in a previous section, Course Request Processing Module.

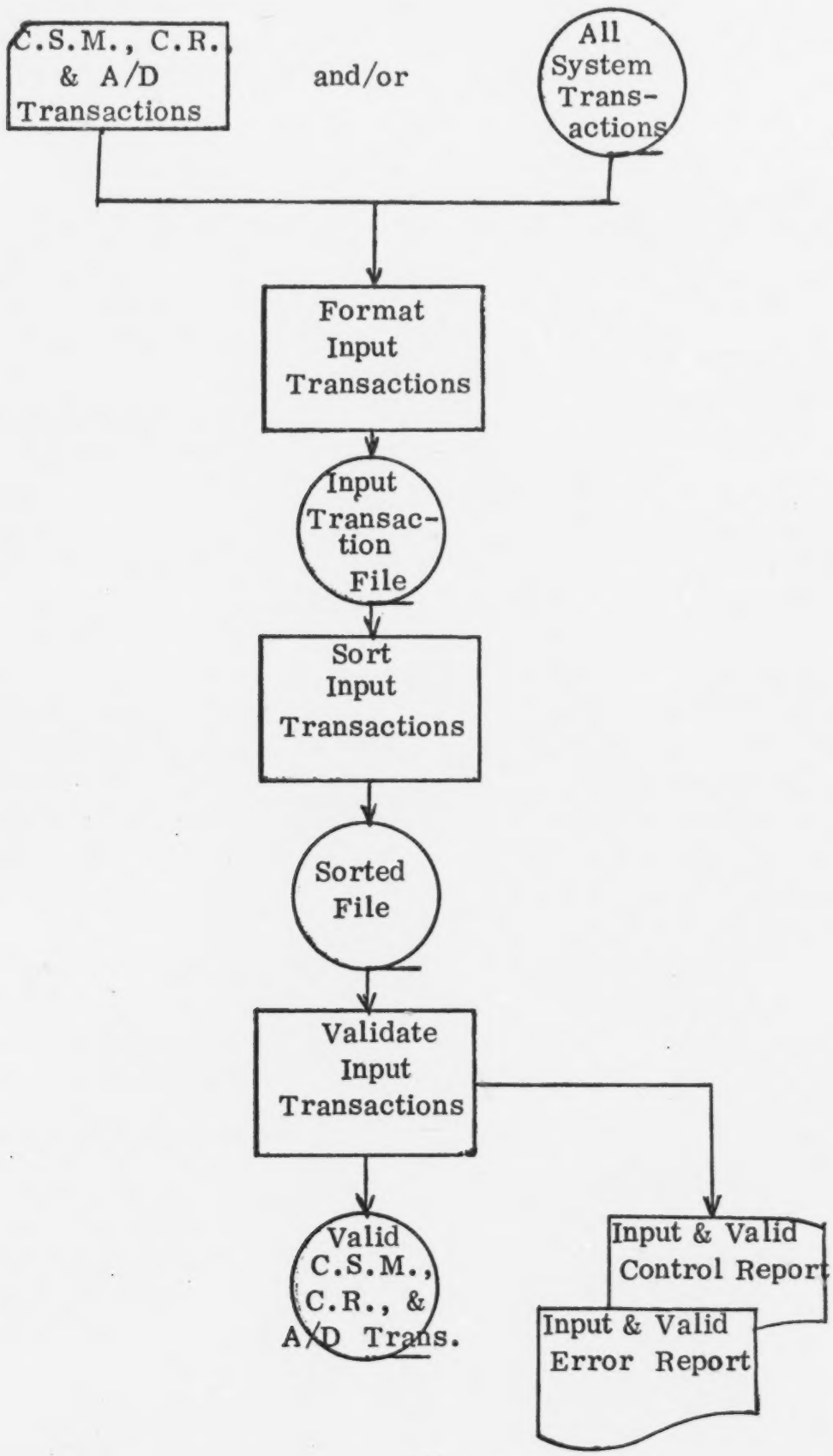
The Add/Drop Module is designed to be run repeatedly during the add/drop process. Each running of the module produces, via parameter card specification, updated versions of the Section List, Section Tally, and student schedules. By parameter card option, the program prints no schedules, schedules for only those students who had an update transaction and therefore changes, or new schedules for all students. When all add/drop processing is completed, final class lists and the latest version of the enrollment file are produced.

The Add/Drop Module may also be used to support registration for late registrants via an "arena style" registration process. Late registrants submit add transactions, approved by appropriate faculty or administrators, for each course for which they wish to register. These transactions are processed, along with transactions which modify pre-existing schedules, to produce initial schedules for late registrants.

The registration and scheduling cycle has been completed with the final add/drop processing. The system requirements as set forth to Colorado State University have been met in the design and implementation of the system.

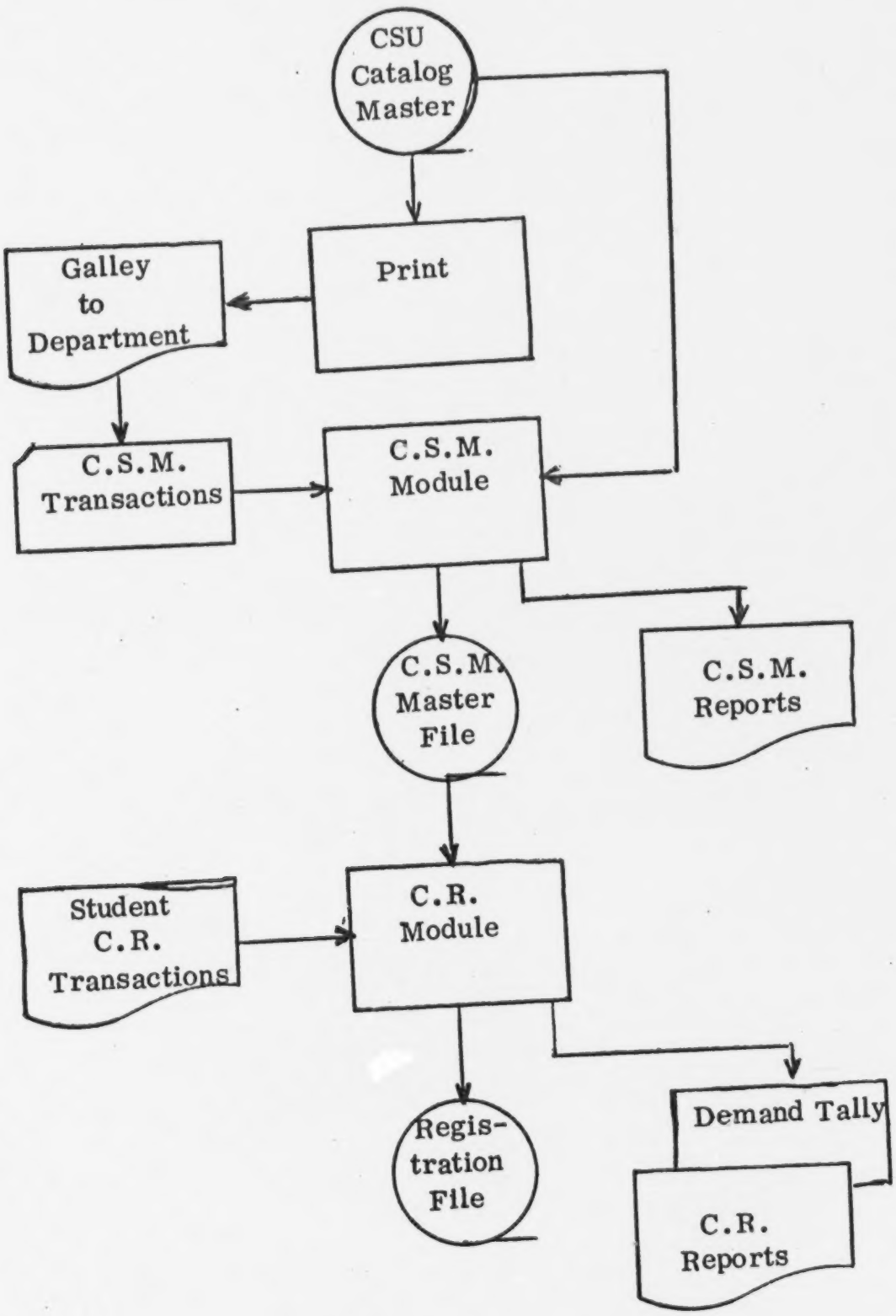
Colorado State University

Transaction Processing for Course Schedule Maintenance, Course Request Processing and Add/Drop Processing Modules



Colorado State University

Course Schedule Maintenance and Course Request Processing Modules



Colorado State University

Student Scheduling and Add/Drop Processing Modules